This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A method for management of a data store comprising:

organizing the data store such that the data store comprises:

~~at least one of~~ an Item;,

an Element;~~,~~ ~~and~~

a plurality of Relationships comprising containment Relationships and reference Relationships, wherein the containment Relationships control the life-time of the Item and are further classified as a holding Relationship or an embedding Relationship, and wherein the Item is deleted when the holding relationship is deleted,

wherein [[:]] said Item is a unit of data storable in a data store and has at least one of a type or subtype, and further ~~comprises~~ includes said Element and said plurality of Relationships~~;~~, and wherein said Element is an instance of a type comprising one or more fields, ~~;~~ and wherein said plurality of Relationships ~~is~~ are a link between at least two Items;

a Base Schema that establishes a framework for creating and organizing each Item and setting the foundational set of properties that the Item may possess; and

a Core Schema that defines a set of core types, wherein each Item is characterized into at least one core type based on the Item type or the Item subtype, the characterizations being stored in the data store.

2. (Previously Presented) The method of claim 1 further comprising organizing the data store such that the data store comprises a plurality of Items, said plurality of Items comprising an Item Folder and at least one other Item that is a member of said Item Folder.

3. (Previously Presented) The method of claim 1 further comprising organizing the data store such that the data store comprises a plurality of Items, said plurality of Items comprising a Category and at least other one Item that is a member of said Category.

4. (Previously Presented) The method of claim 1 wherein a Relationship between two Items is established automatically by a hardware/software interface system.

5. (Previously Presented) The method of claim 1 wherein said Element is understandable by a hardware/software interface system.

6. (Previously Presented) The method of claim 1 further comprising organizing the data store such that the data store comprises a second Element, and wherein said Relationship comprises said second Element.

7. (Previously Presented) The method of claim 1 wherein the Core Schema defines a set of Core Items by which a hardware/software interface system understands and directly processes said set of Core Items in a predetermined and predictable way.

8. (Previously Presented) The method of claim 7 wherein each Item from the set of Core Items is derived from a Common Single Base Item.

9. (Previously Presented) The method of claim 7 wherein said Common Single Base Item is a foundational Item in a Base Schema.

10-24. (Cancelled)

25. (Currently Amended) A computer readable storage medium having stored thereon computer executable instructions for performing steps comprising:

organizing the data store such that the data store comprises:

~~at least one of~~ an Item~~,~~;

an Element~~,~~;~~and~~

a plurality of Relationships comprising containment Relationships and reference Relationships, wherein the containment Relationships control the life-time of the Item and are further classified as a holding Relationship or an embedding Relationship, and wherein the Item is deleted when the holding relationship is deleted,

wherein [[:]] said Item is a unit of data storable in a data store and has at least one of a type or subtype, and further ~~comprises~~ includes said Element and said plurality of

Relationships~~:~~, and wherein said Element is an instance of a type comprising one or more fields, ~~;~~ and wherein said plurality of Relationships ~~is~~ are a link between at least two Items;

a Base Schema that establishes a framework for creating and organizing each Item and setting the foundational set of properties that the Item may possess; and

a Core Schema that defines a set of core types, wherein each Item is characterized into at least one core type based on the Item type or the Item subtype, the characterizations being stored in the data store.

26. (Previously Presented) The computer readable storage medium of claim 25 further comprising computer executable instructions for organizing the data store such that the data store comprises a plurality of Items, said plurality of Items comprising an Item Folder and at least one other Item that is a member of said Item Folder.

27. (Previously Presented) The computer readable storage medium of claim 25 further comprising computer executable instructions for organizing the data store such that the data store comprises a plurality of Items, said plurality of Items comprising a Category and at least other one Item that is a member of said Category.

28. (Previously Presented) The computer readable storage medium of claim 25 wherein a Relationship between two Items is established automatically by a hardware/software interface system.

29. (Previously Presented) The computer readable storage medium of claim 25 wherein said Element is understandable by a hardware/software interface system.

30. (Previously Presented) The computer readable storage medium of claim 25 further comprising computer executable instructions for organizing the data store such that the data store comprises a second Element, and wherein said Relationship comprises said second Element.

31. (Previously Presented) The computer readable storage medium of claim 25 wherein the Core Schema defines a set of Core Items by which a hardware/software interface system understands and directly processes said set of Core Items in a predetermined and predictable way.

32. (Currently Amended) A computer system comprising:

a processor; and

a memory comprising:

~~at least one of~~ an Item~~,~~;

an Element~~,~~; ~~and~~

a plurality of Relationships comprising containment Relationships and reference Relationships, wherein the containment Relationships control the life-time of the Item and are further classified as a holding Relationship or an embedding Relationship, and wherein the Item is deleted when the holding relationship is deleted,

wherein [[:]] said Item is a unit of data storable in a data store and has at least one of a type or subtype, and further ~~comprises~~ includes said Element and said plurality of Relationships~~;~~, and wherein said Element is an instance of a type comprising one or more fields, ~~;~~ and wherein said plurality of Relationships ~~is~~ are a link between at least two Items;

a Base Schema that establishes a framework for creating and organizing each Item and setting the foundational set of properties that the Item may possess; and

a Core Schema that defines a set of core types, wherein each Item is characterized into at least one core type based on the Item type or the Item subtype, the characterizations being stored in the data store.

33. (Previously Presented) The computer system of claim 32 wherein the memory comprises a plurality of Items, said plurality of Items comprising an Item Folder and at least one other Item that is a member of said Item Folder.

34. (Previously Presented) The computer system of claim 32 wherein the memory comprises a plurality of Items, said plurality of Items comprising a Category and at least other one Item that is a member of said Category.

35. (Previously Presented) The computer system of claim 32 wherein a Relationship between two Items is established automatically by a hardware/software interface system.

36. (Previously Presented) The computer system of claim 32 wherein said Element is understandable by a hardware/software interface system.

37. (Previously Presented) The computer system of claim 32 wherein the memory comprises a second Element, and wherein said Relationship comprises said second Element.

38. (Previously Presented) The computer system of claim 32 wherein the Core Schema defines a set of Core Items by which a hardware/software interface system understands and directly processes said set of Core Items in a predetermined and predictable way.